AD-A062 175 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 12/2

AN EVALUATION OF CUSTOMER DELAY IN A TELECOMMUNICATIONS SWITCHB--ETC(U)

SEP 78 J F JENNINGS

UNCLASSIFIED NL

1 OF 1

AD A062175

END

DATE FILMED

3 --79

DDC

IFIED

1 OF 1

AD
A062175

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(2) LEVEL II

NW

NAVAL POSTGRADUATE SCHOOL

Monterey, California

(9) Master's thesis,

THESIS

(6) AN EVALUATION OF CUSTOMER DELAY IN
A TELECOMMUNICATIONS SWITCHBOARD
SUBJECT TO TWO TYPES OF CUSTOMER DEMANDS,

by

(10) Joseph Fulton/Jennings Jr.

(12) 74p.

(11) Sep[redacted]1978

Thesis Advisor: D. P. Gaver

Approved for public release; distribution unlimited.

JOB

251 450

78 12 11 169

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Evaluation of Customer Delay in a a Telecommunications Switchboard Subject to Two Types of Customer Demands		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Joseph Fulton Jennings Jr.		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1978
		13. NUMBER OF PAGES 73
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Multi-server operator switchboard

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

e

A queuing system is investiaged in which two types of customers, Type 1 and Type 2, attempt to access a multi-server operator switchboard. Customer arrivals for each type are assumed to be in accordance with independent Poisson processes with time constant parameters. Service times are assumed to be exponentially distributed and each type of customer has a different service rate. When a queue is formed, Type 1

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601 |

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(20. ABSTRACT Continued)

customers are allowed to queue regardless of the length of the queue but Type 2 customers are blocked when the queue size exceeds a predesignated limit.

An exact solution of the problem was not obtained and a discussion of the complications which precluded an exact solution is presented. A simulation algorithm is developed and validated by comparison with an exact solution of the special case of the problem in which the service rates are equal.

Finally, an analytical approximation is developed and shown to give reasonable results by comparison with the simulation model.

An Evaluation of Customer Delay in A Telecommunications Switchboard Subject to Two Types of Customer Demands

by

Joseph Fulton Jennings Jr.
Captain, United States Marine Corps
B.S., University of Virginia, 1972

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1978

Author: Joseph F. Jennings Jr.

Approved by: Donald P. Gaver
Thesis Advisor

Paul R. Milch
Second Reader

M G Sovereign (J K Hartman)
Chairman, Department of Operations Research

DA Schrady
Dean of Information and Policy Sciences

ABSTRACT

A queuing system is investigated in which two types of customers, Type 1 and Type 2, attempt to access a multi-server operator switchboard. Customer arrivals for each type are assumed to be in accordance with independent Poisson processes with time constant parameters. Service times are assumed to be exponentially distributed and each type of customer has a different service rate. When a queue is formed, Type 1 customers are allowed to queue regardless of the length of the queue but Type 2 customers are blocked when the queue size exceeds a predesignated limit.

An exact solution of the problem was not obtained and a discussion of the complications which precluded an exact solution is presented. A simulation algorithm is developed and validated by comparison with an exact solution of the special case of the problem in which the service rates are equal.

Finally, an analytical approximation is developed and shown to give reasonable results by comparison with the simulation model.

TABLE OF CONTENTS

I. INTRODUCTION

A. STATEMENT OF THE PROBLEM

The problem addressed in this thesis arises in telecommunications systems and concerns the flow of, and delays to, customer traffic through a multiple operator-serviced switchboard. There are two types of customers, Type 1 and Type 2, and each type has its own arrival and service parameters. The arrival and service processes of the two types are statistically independent. All switchboard operators can service either type of customer, and the queuing discipline is first-in, first-out (FIFO). What makes this problem interesting is that while Type 1 customers are allowed to queue regardless of the queue size, Type 2 customers are blocked from joining the queue when the queue length equals or exceeds a predesignated limit.

An example might be useful to ensure that the problem is understood. Consider the civil-sector telephone system, and let the Type 1 customers be those who are direct-dialing a collect, long-distance call, and the Type 2 customers be those who wish the operator to dial the call for them. The two types can be differentiated by the system on the basis of the number of digits they dial; the Type 1 customers would dial a total of eleven digits while the Type 2 customers would dial the single digit 0. Since the Type 1 customers require less work on the part of the operator, it may be

desirable under conditions of heavy use when the queue length gets large, to discourage customers from having the operator dial their calls for them by blocking Type 2 customers from joining the queue. In this case, both types of customers would be allowed to queue when the queue length is small, perhaps receiving a recording advising them that the operators are busy and asking them to wait. However, when the queue length exceeds its predesignated limit, Type 2 customers receive a busy signal or possibly a recording advising them to direct-dial their call.

1. <u>Simplifications and Assumptions</u>

In order to make the problem tractable, a number of complications which would exist in a real-world system have been ignored and a number of simplifying assumptions have been made. Among those elements of the problem which are not being considered are the following:

a. Preemption by a high priority class of customers. A particular example would be customers who dial a three-digit code in the event of an emergency and who must receive preferential service.

b. Balking or reneging. For purposes of this study, it is assumed that once a customer has joined the queue, he remains there until his service commences.

c. The existence of a retrial population. In an actual situation, a Type 2 customer who is blocked from queuing would be expected to do one of three things: (i) decide not to attempt to complete the call at this time; (ii) try accessing

the system as a Type 1 customer; or (iii) continue trying to access the system as a Type 2 customer at repeated, random intervals. The repeated attempts to access the system by those Type 2 customers who exercise option (iii) can be seen to have an effect on the arrival rate, and these customers are said to belong to the retrial population. Although there exist a number of ways to model this phenomena, no attempt has been made to do so in this study. In addition, the increase in the arrival rate of Type 1 customers caused by those Type 2 customers who choose option (ii) is ignored.

The assumptions which were considered necessary were:

d. Arrivals are in accordance with a Poisson process and the arrival rates are constant over time for each type of customer. This is, admittedly, an unrealistic assumption since it is unlikely that the customer arrival rate will be the same at two A.M. as it is at two P.M. Although arrivals according to an inhomogeneous Poisson process can be modeled fairly well by computer simulation, the introduction of time-dependent arrival rates greatly decreases the chances of finding an analytical solution or approximation and it is for this reason that they have not been considered.

e. Service times are exponentially distributed and service rates are constant. This assumption is somewhat less unrealistic since it seems plausible that service times should be the same, on the average, regardless of the time of day.

2. Implications of Unequal Service Rates and the Queuing Discipline

The facts that the two types of customers have different service rates and that Type 2 customers are blocked from the queue when it becomes too long can be seen to have several important implications for the way in which the system functions. In particular, consider the following:

a. Since the service rates are unequal, the system service rate is dependent not only on the number of customers in service, but also on their types. Therefore, any solution technique, either analytical or simulation, must be able to keep track of the composition, by type, of the group of customers undergoing service at every time.

b. When a change occurs in the state of the system, the new composition of the group of customers undergoing service depends not only on whether there are customers waiting in the queue, but also on the precise type of the customer at the head of the queue. Therefore, it is necessary to keep track of the composition, by type, of the queue, in detail.

c. Since the blocking of Type 2 customers effectively sets their arrival rate to zero when the condition occurs, it can be seen that the overall system arrival rate is not constant, but is dependent upon the number in the system. This effect produces an interesting problem in computing the expected waiting time for an arriving customer. This problem is discussed in Chapter Three.

3. Figures of Merit

Three figures of merit were chosen to serve as indicators of system performance. They are: (1) the expected waiting time for a customer of either type who joins the system, (2) the proportion of time that there are no customers of either type in the system and (3) the proportion of time that the queue length equals or exceeds its limit and Type 2 customers are blocked. For all three of the above, only the steady state or long-run value is considered.

B. ANALYTICAL PROCEDURE

1. The Birth-Death Process

The first attempt at a solution was to model the system as a multi-dimensional, birth-death process and then attempt to solve the resulting balance equations algebraically. The necessity of keeping track of the composition, by type, of both the group of customers undergoing service and the group of customers in the queue meant that the minimum number of dimensions which could be used to adequately model the state of the system was four. That is, the state variable which represented the total number of customers in the system would have four dimensions; one each for, the number of Type 1 customers undergoing service, the number of Type 2 customers undergoing service, the number of Type 1 customers in the queue and the number of Type 2 customers in the queue. However, even using a four dimensional state variable proved to

be inadequate to differentiate between the system state in which a Type 1 customer was first in the queue and the state in which a Type 2 customer was the first in the queue. Since the way in which the system changes states is dependent upon the type of the customer at the head of the queue, it would have been necessary to expand the state variable beyond four dimensions to model this effect. The complexity of the balance equations in the four dimensional case offered little hope of finding a solution and an expansion beyond four dimensions would reduce the likelihood of solving the equations still further. For this reason, the multi-dimensional, birth-death process approach was abandoned and other alternatives were considered.

2. Simulation

The next attempt to model the system was via computer simulation since it was hoped that the development and execution of the simulation model would provide insights into the workings of the system which would lead to ideas for analytical solutions or approximations. The development of the simulation model is covered in detail in Chapter two.

3. Verification of the Simulation Model

Although verification of a simulation model is a normal step in the development of the model, it is mentioned separately here for several reasons. First, it was a major effort in itself; secondly, it utilized a generalization of Little's equation for the waiting time that might be of some

interest; and finally, it served as the basis for the first attempt at an analytical approximation. The verification of the simulation is covered in detail in Chapter Three.

4. <u>An Analytical Approximation</u>

Chapter Four details the derivation of an approximate analytical solution based on the equations developed in Chapter Three for verifying the simulation model. In addition, the rationale for developing an analytical approximation model and the advantages to be gained from using such a model are presented. Finally, comparisons are made between results obtained from the analytical approximation and the simulation to determine the degree of agreement between the two models.

II. THE SIMULATION ALGORITHM

A. RATIONALE FOR THE SIMULATION APPROACH

Although it is generally accepted that an analytical solution is preferable to a simulation, there are a number of situations in which a simulation approach is indicated. Foremost among these is the case in which an analytical solution does not exist or, equivalently, an analytical solution exists in theory, but is unobtainable in practice. As was mentioned in Chapter One, the latter situation is the case in the present problem. If it is impractical to obtain a complete and useable analytical solution, a search for a simple analytical approximation is likely to be worthwhile. Given the complexities of the problem of this thesis, derivation of an analytical approximation becomes a non-trivial task. Therefore, the simulation approach is utilized to gain insight into the problem and the operation of the queuing system in order to facilitate later analytical work.

Another reason for simulating the system is that if an analytical approximation can be found, then the simulation can serve as a basis of comparison. This, of course, assumes a high degree of confidence in the simulation model which means that the simulation must somehow be verified. This important step in the development of the simulation model is covered in Chapter Three.

Finally, if a suitable analytical approximation cannot be found, then the simulation remains the only tool with which to study the system.

B. CONCEPTUAL OPERATION OF THE SYSTEM

The first step in the development of the simulation algorithm was to study the operation of the system so as to make sure that the manner in which the system changed from one state to another was fully understood. The system was viewed as staying in a current state for a random time after which one of four statistically independent events would occur with certain specified probabilities. The occurrence of these events causes the system to change state. The four possible events are: (i) a Type 1 customer arrives and joins the system, (ii) a Type 2 customer arrives and may or may not join the system depending on the current state, (iii) a Type 1 customer completes service, or (iv) a Type 2 customer completes service. The conceptual operation of the system and the four events along with their probabilities of occurrence are displayed in Figure 2.1. Figure 2.2 shows the system changes associated with the occurrence of each of the events. Terms used in the two figures are defined below.

J1 = number of Type 1 customers in service.

J2 = number of Type 2 customers in service.

Q1 = number of Type 1 customers in the queue.

Q2 = number of Type 2 customers in the queue.

S = number of servers.

T = queue threshold (applies to Type 2 customers).

N = total number of customers in the system
= J1 + J2 + Q1 + Q2

λ_1 = Type 1 arrival rate.

λ_2 = Type 2 arrival rate.

μ_1 = Type 1 service rate.

μ_2 = Type 2 service rate.

d = $\lambda_1 + \lambda_2 + J1\mu_1 + J2\mu_2$

A few notes will be helpful to fully understand the two figures.

1. The "sojourn time" is the random time period during which the system stays in its current state. The sojourn time is exponentially distributed with mean equal to $\frac{1}{\lambda_1 + \lambda_2 + J1\mu_1 + J2\mu_2}$ which is $\frac{1}{d}$. Note that this means that the distribution of the sojourn time is not fixed, but that it changes with each change in J1 and/or J2.

2. According to the queuing discipline assumed for this problem, a Type 2 customer can only join the system when N, the number in the system, is less than T, the queue threshold. This is illustrated in Figure 2.2, where it can be seen that no state change takes place if a Type 2 arrival occurs when the number in the system is larger than the cutoff value.

3. Figure 2.2 also illustrates the fact that when a queue is formed, the new state of the system depends upon whether a Type 1 or Type 2 customer is at the head of the queue.

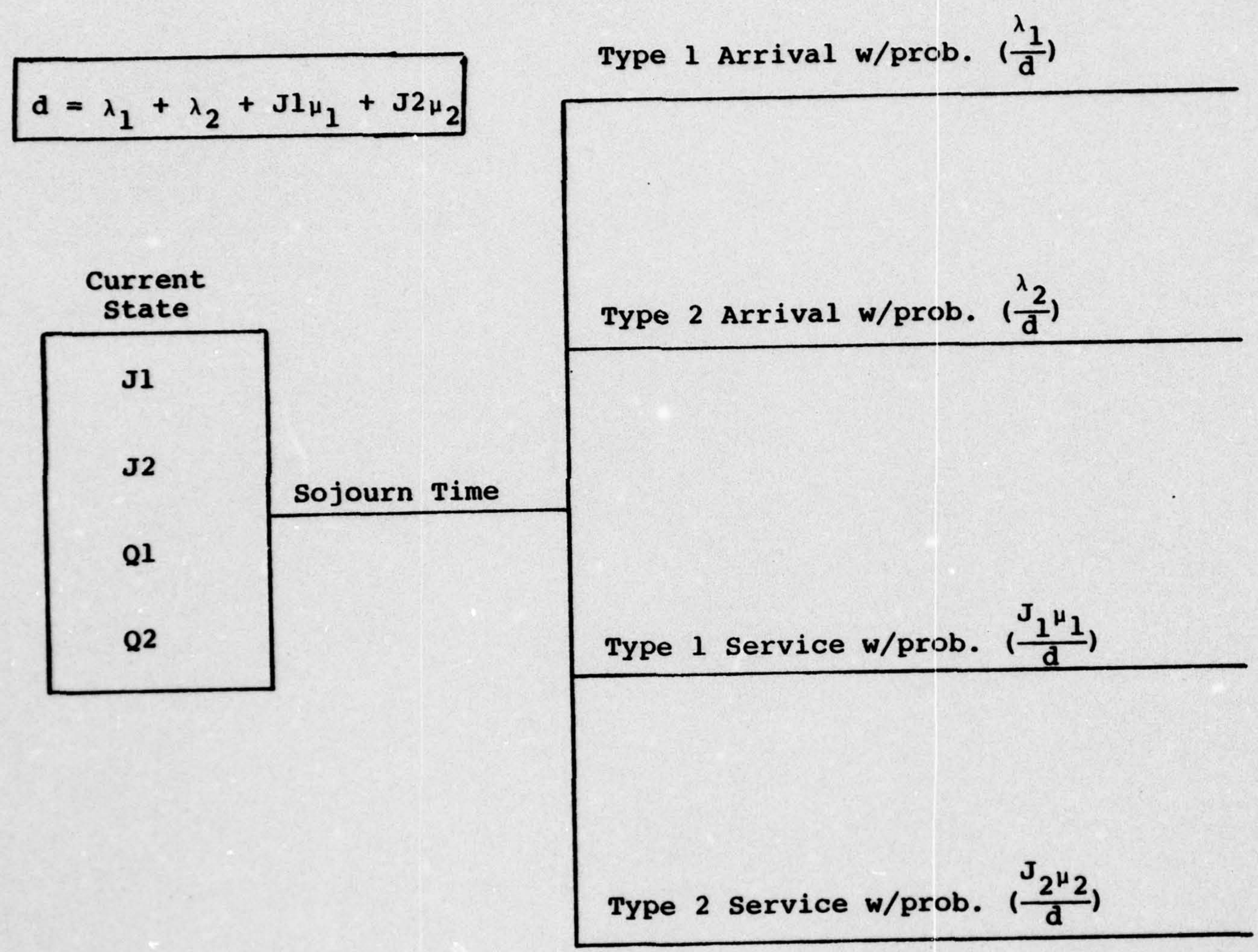


FIGURE 2.1. Possible Events and Their Probabilities of Occurrence

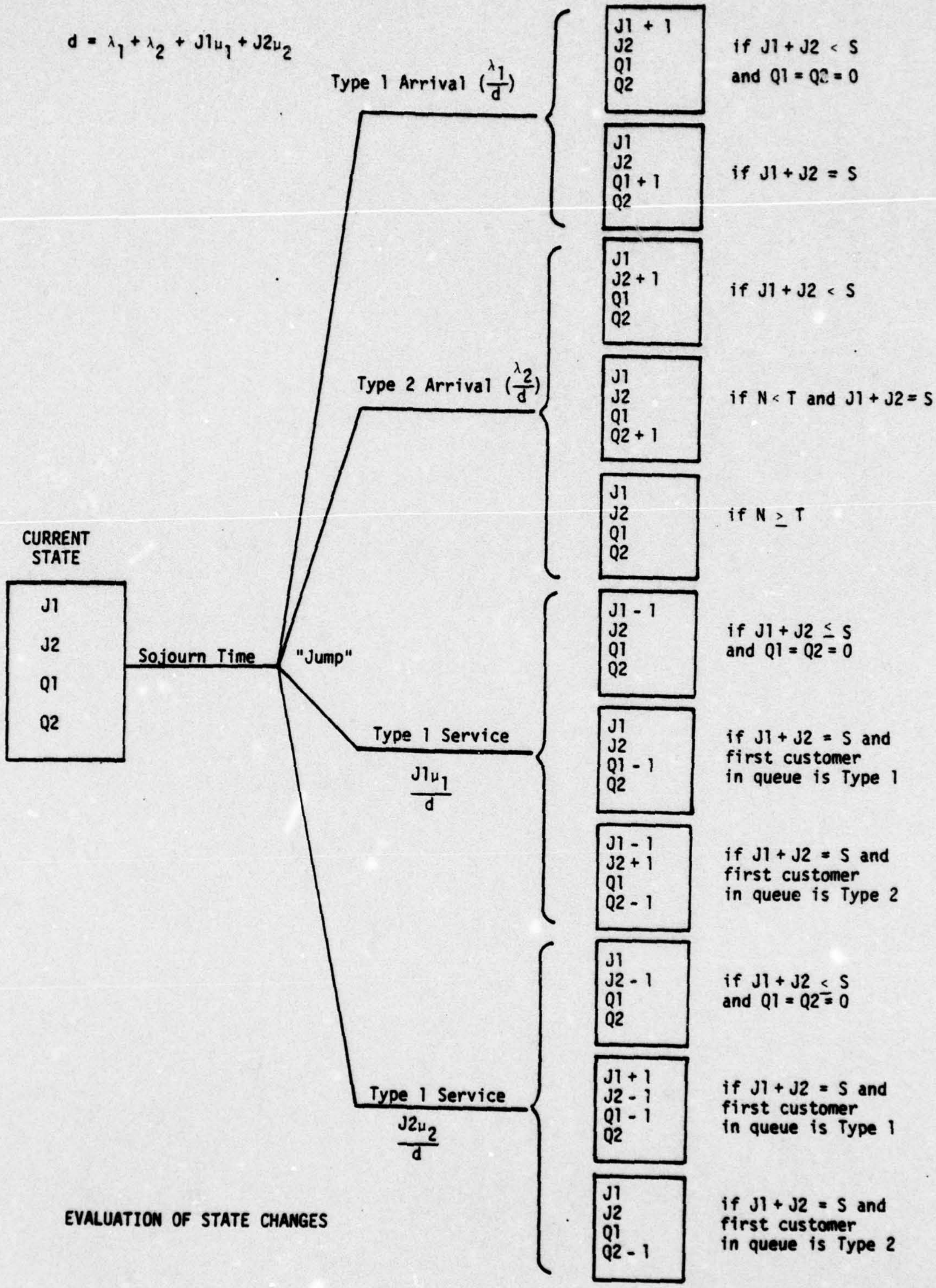

d = λ1 + λ2 + J1μ1 + J2μ2
CURRENT STATE
J1 J2 Q1 Q2
Sojourn Time
"Jump"
Type 1 Arrival (λ1/d)
Type 2 Arrival (λ2/d)
Type 1 Service J1μ1/d
Type 1 Service J2μ2/d
J1 + 1 J2 Q1 Q2
if J1 + J2 < S and Q1 = Q2 = 0
J1 J2 Q1 + 1 Q2
if J1 + J2 = S
J1 J2 + 1 Q1 Q2
if J1 + J2 < S
J1 J2 Q1 Q2 + 1
if N < T and J1 + J2 = S
J1 J2 Q1 Q2
if N ≥ T
J1 - 1 J2 Q1 Q2
if J1 + J2 ≤ S and Q1 = Q2 = 0
J1 J2 Q1 - 1 Q2
if J1 + J2 = S and first customer in queue is Type 1
J1 - 1 J2 + 1 Q1 Q2 - 1
if J1 + J2 = S and first customer in queue is Type 2
J1 J2 - 1 Q1 Q2
if J1 + J2 < S and Q1 = Q2 = 0
J1 + 1 J2 - 1 Q1 - 1 Q2
if J1 + J2 = S and first customer in queue is Type 1
J1 J2 Q1 Q2 - 1
if J1 + J2 = S and first customer in queue is Type 2
EVALUATION OF STATE CHANGES

FIGURE 2.2

C. STRUCTURE OF THE ALGORITHM

After the operation of the system was understood and represented by a probability model, the next step was to write out a simulation algorithm in flow-chart form. The algorithm must be capable of performing at least the following functions: (i) it must automatically choose an appropriate value for each sojourn time based on its <u>current</u> probability distribution, (ii) it must determine which of the four possible events is to occur next, (iii) it must update the state of the system based on the type of event that occurred, the value of J1, J2, Q1, Q2, and the type of customer at the head of the queue, if formed, and finally (iv) it must gather appropriate statistics on the figures of merit. The general flow-chart of Figure 2.3 describes the way in which the program performs these four functions. Notice that the algorithm essentially consists of one loop embedded in another. The inner loop performs the major tasks of choosing events and sojourn times and updating the system variables. This loop continues to iterate for a specified length of simulation time, for example one hour. When the time limit is reached, control is passed to the outer DO loop which gathers interim statistics and then checks to see if an iteration counter has reached its maximum value. If it has not, then control is passed back to the inner loop and the process starts anew with the values of the system variables preserved from the end of the previous iteration.

GENERAL FLOW CHART FOR SIMULATION ALGORITHM

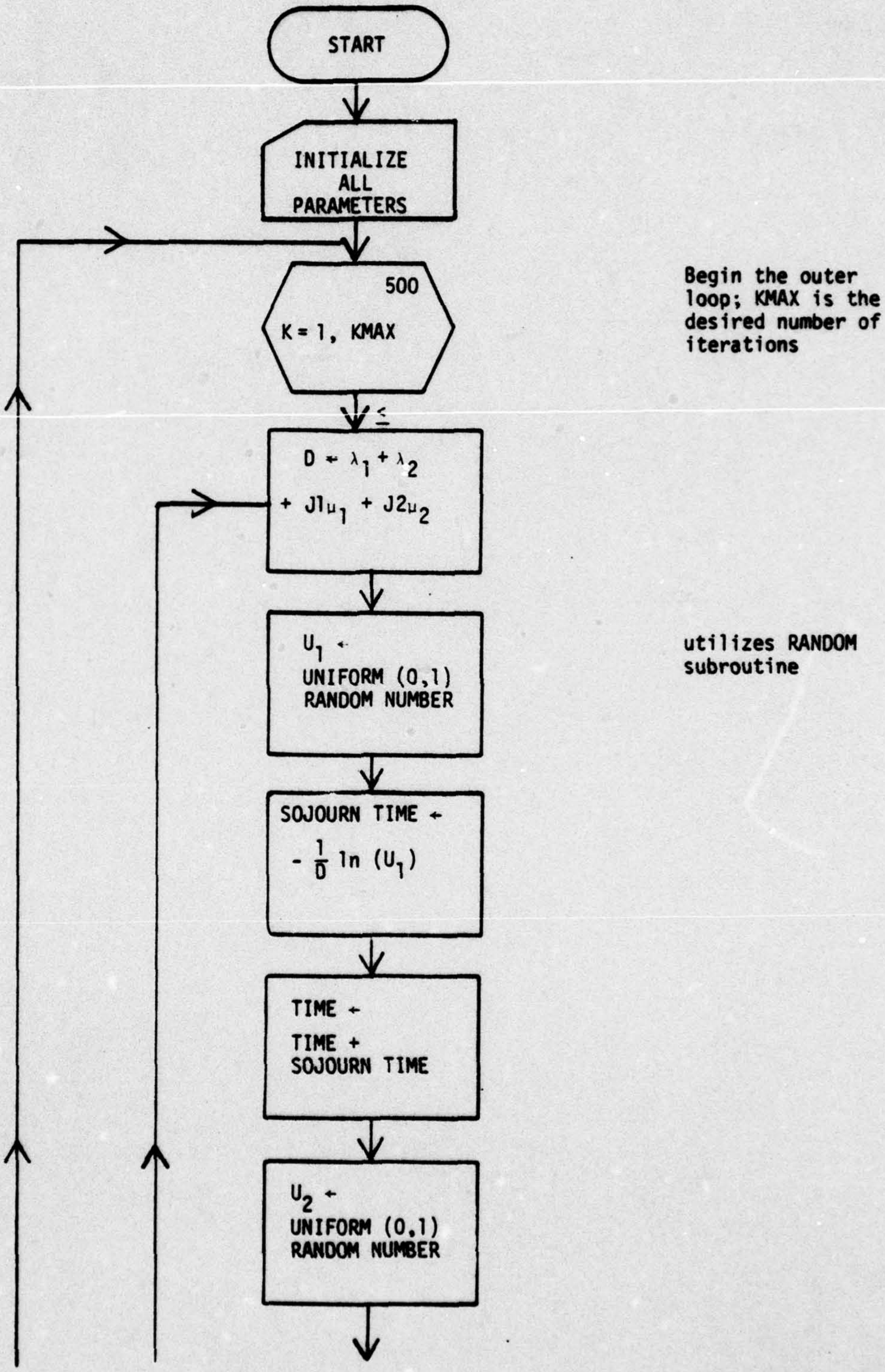
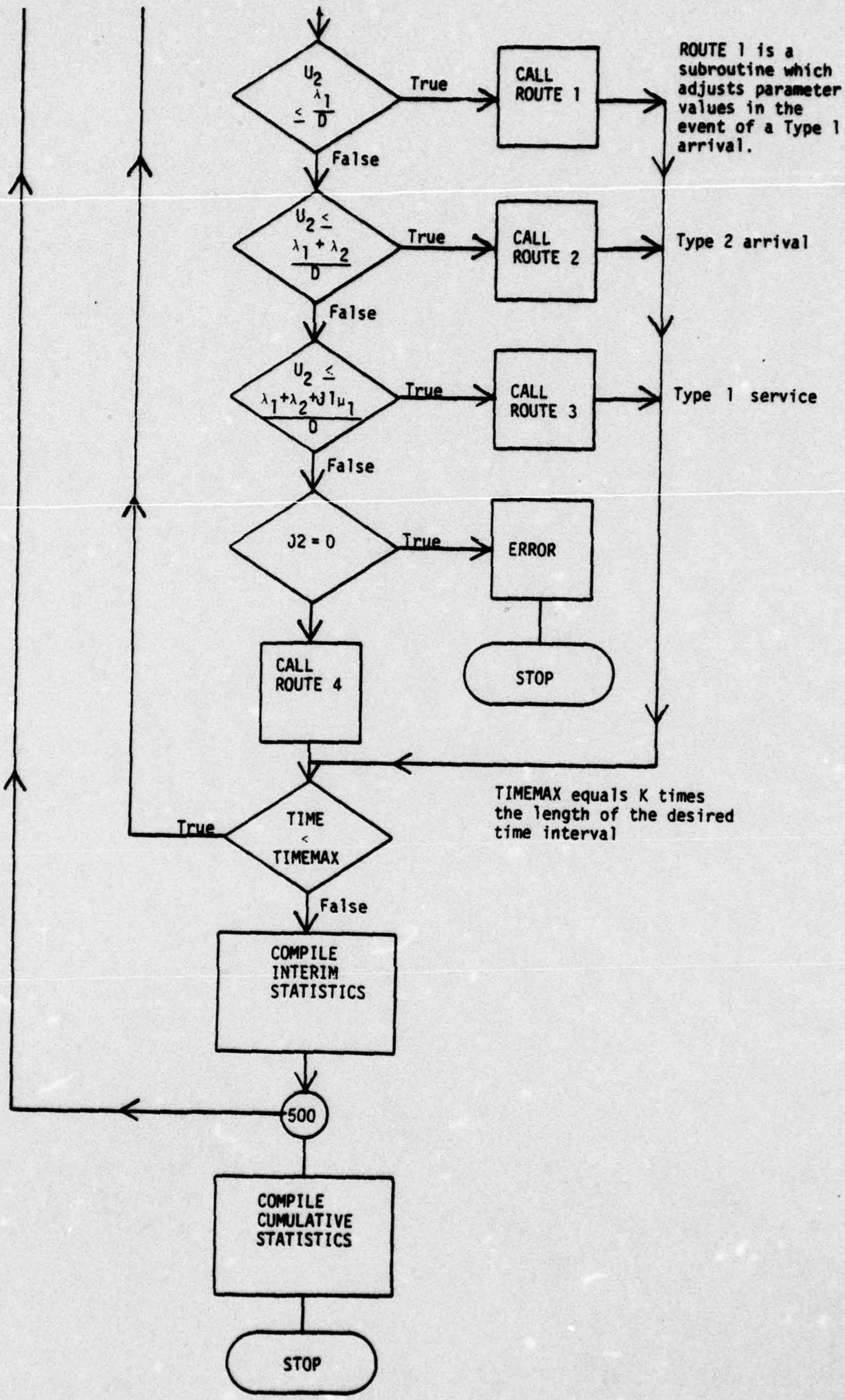


FIGURE 2.3



$U_2 \le \frac{\lambda_1}{D}$
True
False
CALL ROUTE 1
ROUTE 1 is a subroutine which adjusts parameter values in the event of a Type 1 arrival.
$U_2 \le \frac{\lambda_1 + \lambda_2}{D}$
True
False
CALL ROUTE 2
Type 2 arrival
$U_2 \le \frac{\lambda_1 + \lambda_2 + J1\mu_1}{D}$
True
False
CALL ROUTE 3
Type 1 service
J2 = 0
True
ERROR
STOP
CALL ROUTE 4
TIME < TIMEMAX
True
False
TIMEMAX equals K times the length of the desired time interval
COMPILE INTERIM STATISTICS
500
COMPILE CUMULATIVE STATISTICS
STOP

If the iteration counter has reached its upper limit, then final statistics are gathered and the algorithm terminates. In this manner one day of system operation can be simulated by setting the upper limit for the inner loop to one hour and the upper limit for the outer loop to twenty-four. The algorithm uses four sub-algorithms called ROUTE1, ROUTE2, ROUTE3 and ROUTE4. The sub-algorithms are utilized when an event occurs and their function is to correctly update the system variables according to the rules depicted in Figure 2.2. Therefore, for instance, ROUTE2, which is utilized when a Type 2 arrival occurs, would first check whether all the servers were busy; if not, it would increase the number of Type 2 customers in service by one. If the servers were all busy, ROUTE2 would then test whether the number in the system equaled or exceeded the queue threshold; if not then the number of Type 2 customers in the queue would be increased by one. If the number in the system were greater than the queue threshold, there would be no change in the state of the system and control would be passed back to the main algorithm. The flow-chart for ROUTE2 is shown in Figure 2.4. The flow-charts for the other sub-algorithms are similar.

The composition of the queue is maintained in a vector of one's and two's, called NEXT1, in which a one in the i^{th} position indicates that the i^{th} customer in the queue is a Type 1. Each time a customer arrives, another element is added to the vector; either a one or a two depending on

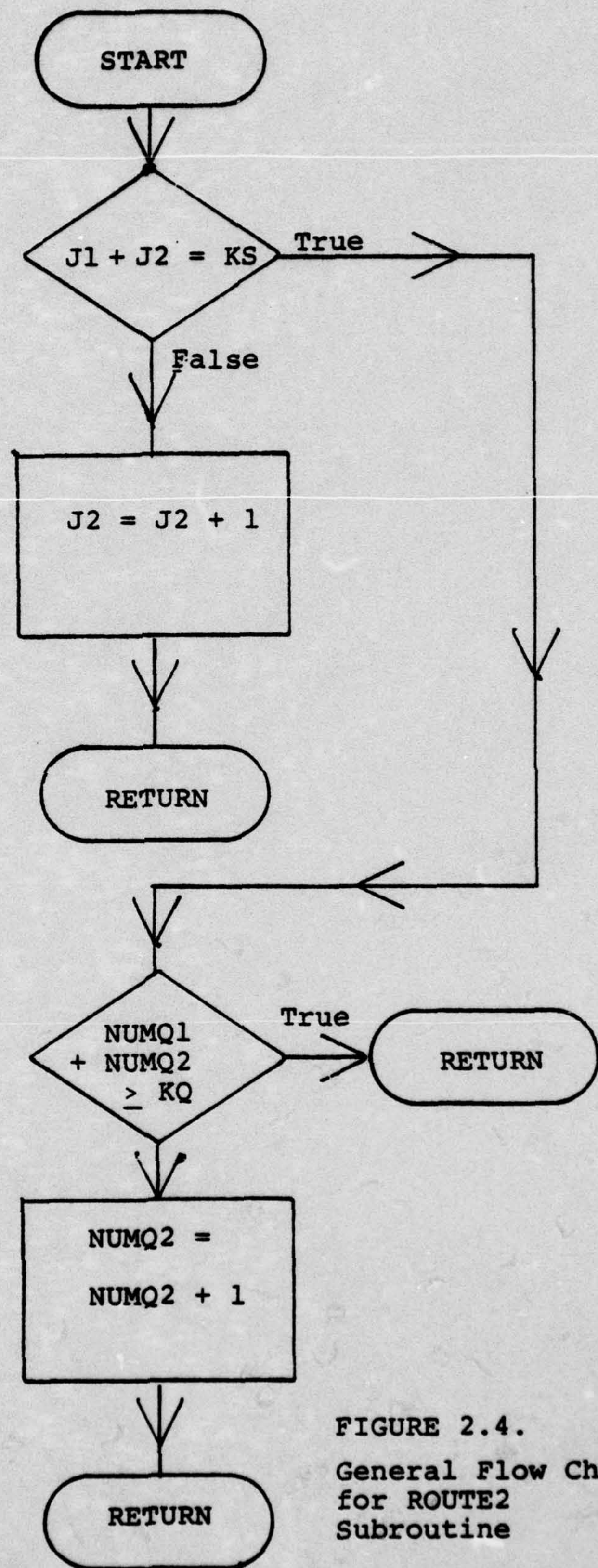


FIGURE 2.4.

General Flow Chart for ROUTE2 Subroutine

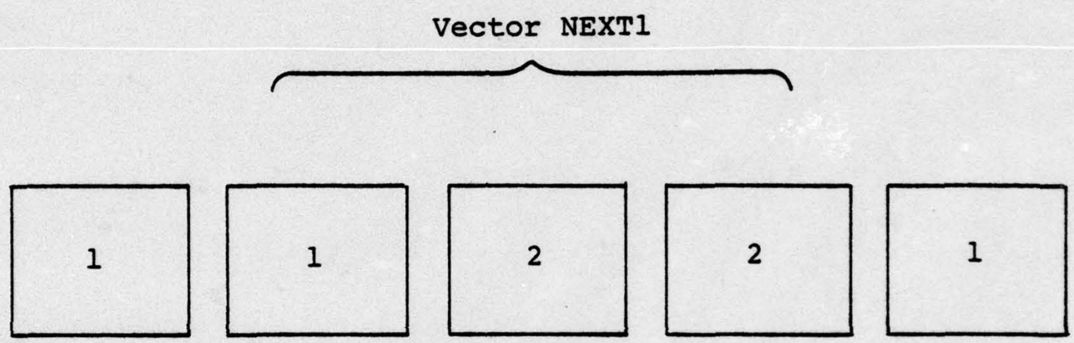
the type of customer who arrived. Each time a service event occurs, a counter is incremented by one. When this counter is used as the subscript of NEXT1, it points to the element in the vector which represents the customer currently at the head of the queue. For example, assume that five arrival events have occurred and that the composition of the vector NEXT1 is as shown in Figure 2.5. Let the value of the service counter be four indicating that the fourth service event is about to occur. Then NEXT1(4) indicates that the customer at the head of the queue is a Type 2.

Once the problem was understood and the flow-chart written, the next step was to choose an appropriate computer language and program the algorithm.

D. THE COMPUTER PROGRAM

Fortran IV was chosen as the language to implement the simulation algorithm. Initially, consideration was given to simulation-oriented languages such as SIMSCRIPT and GPSS, but it was felt that their sophistication and abstraction would tend to obscure some of the inner workings of the queuing system and would thus offer less insight into the problem. Coding of the program was a straightforward translation of the flow-chart. A program listing, with explanatory comments and a sample output, is provided at the end of the thesis.

The three figures of merit are printed out after each hour. These are cumulative statistics — that is, they



NEXT1(4) points to the fourth element in NEXT1 and indicates that when the fourth service event occurs, the customer at the head of the queue is Type 2.

FIGURE 2.5. Method of Identifying the Type of the Customer Currently at the Head of the Queue

reflect the operation of the system from the start of the simulation, and not just from the previous hour. The printing out of cumulative statistics at the end of each hour was necessary in order to ensure that the simulation had been allowed to run long enough to reach the practical equivalent of steady-state. Note that there is little fluctuation in the values of the figures of merit toward the end of the program run. If there had continued to be significant fluctuation, it would have indicated that the system was not in steady-state and that the program should be run again for longer simulation time. The program keeps track of the composition of the system, by customer type, at all times and, for each customer who joins the queue, it computes the actual time he waits for service. Thus, the average waiting times are computed by simply totaling all of the waiting times and dividing by the number of customers who joined the queue. The probability that there are no customers in the system is actually the long-run fraction of time that the system was in that state. Similarly, the probability that a Type 2 customer is blocked is the long-run fraction of time that the queue threshold was exceeded.

The program utilizes the IMSL sub-routine RANDOM to generate the pseudo-random numbers needed for determining the sojourn time and choosing the next event. RANDOM takes as input a "seed" number and, for a given seed, will always produce the same string of numbers which are uniformly distributed on the range from zero to one.

Looking at the print-out from the simulation program, the first question that might be asked is: "Are the results meaningful — do they accurately represent the functioning of the real system?" Thus the need for some kind of verification of the simulation program becomes obvious. The process of verification is detailed in the next chapter.

III. VERIFICATION OF THE SIMULATION ALGORITHM

A. NEED FOR VERIFICATION

Verification of a simulation model by testing it against the results obtained from a mathematical model is the final step in its development. As mentioned in Chapter I, the verification process has been separated from the rest of the development of the algorithm in order to better highlight some interesting results. Of course, as is often the case, the reason for going to a simulation approach in the first place was that no general mathematical model existed. However, it was felt that if a mathematical model could be developed for a special case and the simulation verified against it, then it would at least indicate that the simulation was performing properly. This is the approach that was taken, and it is detailed below.

B. METHOD OF VERIFICATION

1. The Mathematical Model

It was decided to attempt to model the special case of the problem in which the service times of the Type 1 and Type 2 customers are equal. All other aspects of the problem remain unchanged; that is, the arrivals occur according to a homogeneous Poisson process with different parameters for each type of customer, and Type 2 customers are blocked from joining the queue when the queue length equals or

exceeds a given threshold. Comparison between the mathematical model and the simulation model was made on the basis of the three figures of merit mentioned in Chapter I; those were, (i) the probability that the system was empty, (ii) the expected waiting time for any customer and (iii) the probability that a Type 2 customer would be blocked. The equations for (i) and (iii) are, although complicated and tedious, straightforward solutions of the balance equations which resulted from application of the theory of birth-death processes.[1] The derivation of these equations is detailed in sections two and four of this chapter.

It appeared that the simplest way of computing the expected waiting time was to utilize Little's equation[2];

$$E[W] = \frac{E[Q]}{\lambda} . \tag{3.1}$$

Where:

$E[W]$ = the expected waiting time of any customer

$E[Q]$ = the expected number of customers in the queue

λ = the system arrival rate

---

[1] Gaver, D.P., and Thompson, G.L., Programming and Probability Models in Operations Research, p. 467-482, Brooks/Cole, 1973.

[2] Hillier, F.S., and Lieberman, G.J., Introduction to Operations Research, p. 291-292, Holden-Day, 1967.

Little's equation, as shown above, assumes a constant arrival rate[3] and is not directly applicable to the problem under study since, as was discussed in Chapter I, the system arrival rate depends upon the number of customers in the system. Therefore, it was decided to modify Little's equation as follows:

$$E[W] = \frac{E[Q]}{\lambda_E} \tag{3.2}$$

Where:

λ_E = the "effective" rate at which customers join the system.

$$\lambda_E = \lambda_1 + \lambda_2 \sum_{n=0}^{T-1} p_n \tag{3.3}$$

p_n = the probability that there are exactly n customers in the system.

Before equation (3.2) can be verified, it will be necessary to set up the balance equations and derive an expression for p_n. At the same time, an expression for the probability that the system is empty will be derived.

---

[3]Hillier and Lieberman, p. 291-292.

2. A Derivation of an Expression for the Probability That the System is Empty

The theory of birth-death processes provides expressions for long-run or steady-state probability that the number of customers in the system is a given value. To determine these probabilities, it is first necessary to define some terms. Let

N = the number of customers in the system,

p_n = the probability that $N = n$,

λ_1 = the arrival rate for Type 1 customers,

λ_2 = the arrival rate for Type 2 customers,

μ = the service rate for both types of customers,

$\lambda^{(i)}$ = overall system arrival rate when $N = i$,

T = queue threshold; that is when $N \geq T$, Type 2 customers are blocked from the queue,

s = the number of servers,

$\mu^{(i)}$ = system service rate when $N = i$.

It can then be shown that an expression for p_n is[4]

$$p_n = \frac{\lambda^{(0)} \cdot \lambda^{(1)} \cdot \lambda^{(2)} \cdots \lambda^{(n-1)}}{\mu^{(1)} \cdot \mu^{(2)} \cdot \mu^{(3)} \cdots \mu^{(n)}} p_0 \tag{3.4}$$

---

[4]Gaver and Thompson, pp. 467-482.

which comes directly from the basic balance equation. This leads to the following expression for p_0:

$$p_0 = \frac{1}{1 + \sum_{i=1}^{\infty} \{ \prod_{j=1}^{i} \frac{\lambda^{(j-1)}}{\mu^{(j)}} \}} \tag{3.5}$$

The remainder of this section is devoted to manipulating equation (3.4) into a form suitable for use with the problem at hand.

The first step is to write down the system arrival and service rates that result from the queuing discipline:

$$\begin{array}{lll}
\lambda^{(0)} = \lambda_1 + \lambda_2 & \qquad & \mu^{(0)} = 0 \\
\lambda^{(1)} = \lambda_1 + \lambda_2 & & \mu^{(1)} = \mu \\
\vdots & & \mu^{(2)} = 2\mu \\
 & & \vdots \\
\lambda^{(T-1)} = \lambda_1 + \lambda_2 & & \\
 & & \mu^{(s)} = s\mu \\
\lambda^{(T)} = \lambda_1 & & \mu^{(s+1)} = s\mu \\
\vdots & & \vdots \\
\lambda^{(n)} = \lambda_1 & & \mu^{(n)} = s\mu \; .
\end{array}$$

Rewriting equation (3.5) in terms of the above expression yields an expression for $_0$ in terms of the system service rate and the rate at which <u>customers</u> <u>actually</u> <u>join</u> <u>the</u> <u>system</u>. This is an important point and it will be repeatedly emphasized in the remainder the the thesis. In a "normal" queuing system, the arrival rate is identical to the rate at which customers join the system. However, a distinction must be made in any system in which the queuing discipline allows customers to be blocked from the system. The correct expression for p_0, in terms of the rate at which customers join the system, is shown below:

$$p_0 = \frac{1}{1 + [\sum_{n=1}^{s} \frac{(\lambda_1+\lambda_2)^n}{n!\mu^n} + \sum_{n=s+1}^{T} \frac{(\lambda_1+\lambda_2)^n}{s!s^{(n-s)}\mu^n} + \sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^T \lambda_1^{(n-T)}}{s!s^{(n-s)}\mu^n}]}$$

Next examine each of the summations in the denominator to see if they can be summed in closed form.

a. $\sum_{n=1}^{s} \frac{(\lambda_1+\lambda_2)^n}{n!\mu^n}$. This expression cannot be summed in closed form; however, the sum can easily be computed in two ways; (i) by direct summation if s is not large or (ii) by reference to a table of cumulative Poisson probabilities. The second method is possible since

$$\frac{((\lambda_1+\lambda_2)/\mu)^n}{n!} \exp(- \frac{\lambda_1+\lambda_2}{\mu})$$

is the density function, with respect to counting measure, for a Poisson distribution with parameter $\frac{\lambda_1+\lambda_2}{\mu}$. Thus if the figure obtained from a cumulative Poisson table look-up with the appropriate parameter is dived by $\exp(-\frac{\lambda_1+\lambda_2}{\mu})$ then the desired sum will result.

b. $\sum_{n=s+1}^{T} \frac{(\lambda_1+\lambda_2)^n}{s!s^{(n-s)}\mu^n}$. This sum can be expressed

in closed form as follows:

$$= \frac{1}{s!}\left[\sum_{n=s+1}^{\infty} \frac{(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n} - \sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n}\right]$$

$$= \frac{1}{s!}\left[\frac{(\lambda_1+\lambda_2)^{s+1}}{s\mu^{s+1}} \sum_{j=0}^{\infty} \left(\frac{(\lambda_1+\lambda_2)}{s\mu}\right)^j\right.$$

$$\left. - \frac{(\lambda_1+\lambda_2)^{(T+1)}}{s^{(T+1-s)}\mu^{(T+1)}} \sum_{j=0}^{\infty} \left(\frac{\lambda_1+\lambda_2}{s\mu}\right)^j\right]$$

$$= \frac{1}{s!\left(1-\frac{\lambda_1+\lambda_2}{s\mu}\right)}\left[\frac{(\lambda_1+\lambda_2)^{(s+1)}}{s\mu^{(s+1)}} - \frac{(\lambda_1+\lambda_2)^{(T+1)}}{s^{(T+1-s)}\mu^{(T+1)}}\right] .$$

c. $\sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^T \lambda_1^{(n-T)}}{s!s^{(n-s)}\mu^n}$. This sum can also be

expressed in closed form:

$$\sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^T \lambda_1^{(n-T)}}{s!\,s^{(n-s)}\mu^n} = \frac{(\lambda_1+\lambda_2)^T}{s!} \sum_{n=T+1}^{\infty} \frac{\lambda_1^{(n-T)}}{s^{(n-s)}\mu^n}$$

$$= \frac{(\lambda_1+\lambda_2)^T}{s!}\left(\frac{\lambda_1}{s^{(T+1-s)}\mu^{(T+1)}}\right)\left(\frac{1}{1-\frac{\lambda_1}{s\mu}}\right)$$

Combining the above sums yields the desired expression for p_0:

$$p_0 = \frac{1}{1+\sum\limits_{n=1}^{s}\frac{(\lambda_1+\lambda_2)^n}{n!\,\mu^n} + \left\{\frac{1}{s!}\left(\frac{1}{1-\frac{\lambda_1+\lambda_2}{s\mu}}\right)\left[\frac{\lambda_1+\lambda_2}{s\mu^{(s+1)}} - \frac{(\lambda_1+\lambda_2)^{(T+1)}}{s^{(T+1-s)}\mu^{(T+1)}}\right]\right\} + \left[\left(\frac{(\lambda_1+\lambda_2)^T}{s!}\right)\left(\frac{\lambda_1}{s^{(T+1-s)}\mu^{T+1}}\right)\left(\frac{1}{1-\frac{\lambda_1}{s\mu}}\right)\right]} \tag{3.6}$$

The next step is to derive equation (3.2), the modified form of Little's equation.

3. <u>Derivation of Little's Equation for the Case Where the Service Rate is a Function of the Number of Customers in the System</u>

The validation of equation (3.2) proceeded in two steps; first, the equation was derived in general to be applicable to any queing system and next, the general equation was modified to yield a form directly applicable

to the problem at hand. Before beginning the derivation, it was necessary to define a number of quantities.

T = the queue threshold.

$\Lambda(n)$ = combined rate at which customers join the system when $N = n$.

A = the event that a customer, of either type, arrives and joins the system.

W = waiting time for any customer.

p_n = probability that $N = n$.

In the general case, the desired form of the expression for the expected waiting time is:

$$E[W] = \frac{\text{total expected waiting time for all customers who join the system during time } t}{\text{total expected number of customers who join the system during time } t.} \tag{3.7}$$

In order to write the above expression in mathematical form, first consider the expected waiting time of a customer, given that a customer arrives and joins the system when $N = n$. This can be seen to equal zero if there are any idle servers and to equal the expected service times of all customers ahead of him in the queue plus the expected service time of the customer being served if the servers are all busy and a queue is formed. This can be expressed as:

$$E[W|A,N=n] = \begin{cases} 0 & \text{if } n < s \\ \frac{n-s+1}{s} & \text{if } n \geq s \end{cases}$$

Now consider the expected contribution to the total expected steady-state waiting time which results from a customer arrival in a short time interval (dt). This is simply the expression derived above, multiplied by the steady-state probability that a customer joins the system during a time interval dt when there are n customers in the system ($\Lambda(n)dt$), and by the probability that there are n customers in the system (p_n), and then summed over the range of N. The expression appears as:

$$\sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu} \Lambda(n) p_n\} \, dt \quad .$$

If this last expression is integrated over the range (0,t) then, in the limit as t goes to infinity, the result is equal to the total expected waiting time for all customers who joined the system during time t, which is the numerator of equation (3.7). That is:

$$\int_0^t \sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu}\Lambda(n) p_n\} dt \approx \sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu} \Lambda(n) p_n\} t \quad .$$

The denominator of equation (3.7) is simply:

$$\sum_{n=0}^{\infty} (\Lambda(n)p_n)t \;.$$

Combining the expression for the numerator and denominator yields the desired expression for the expected waiting time which is applicable in general:

$$E[W] \;=\; \frac{\sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu}\Lambda(n)p_n\}}{\sum_{n=0}^{\infty} \Lambda(n)p_n}$$

Equation (3.8) can be used to determine the expected waiting time at steady-state for any queuing system in which the rate at which customers join the system is a function of the number of customers in the system. In order to make equation (3.8) more directly applicable to the particular system under study, it is necessary to define some more terms. They are:

$\omega_1(n)$ = the probability that a Type 1 customer can join the system upon arrival when there are n customers in the system.

$\omega_2(n)$ = the probability that a Type 2 customer can join the system upon arrival when there are n customers in the system.

Note again the important distinction made between the rate at which customers arrive (λ_1 and λ_2) and the rate at which customers joint the system by either join directly into service or joining the queue ($\Lambda(n)$). The two rates are related as follows:

$$\Lambda(n) = \lambda_1\omega_1(n) + \lambda_2\omega_2(n) \tag{3.9}$$

Combining equations (3.8) and (3.9) yields the desired expression for the expected waiting time.

$$E[W] = \frac{\lambda_1 \sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu} \omega_1(n)p_n\} + \lambda_2 \sum_{n=s}^{\infty} \{\frac{n-s+1}{s\mu}\omega_2(n)p_n\}}{\lambda_1 \sum_{n=0}^{\infty} \omega_1(n)p_n + \lambda_2 \sum_{n=0}^{\infty} \omega_2(n)p_n} \tag{3.10}$$

Equation (3.10) can now be applied directly to the problem of interest by inserting appropriate values for $\lambda_1, \lambda_2, \omega_1(n)$ and $\omega_2(n)$. However, in order to display the connection between equation (3.10) and equation (3.2), the modified form of Little's equation, some additional work is necessary.

Recall equation (3.4) in section two of this chapter:

$$p_n = \frac{\lambda^{(0)} \cdot \lambda^{(1)} \cdot \lambda^{(2)} \ \ldots \ \lambda^{(n-1)}}{\mu^{(1)} \cdot \mu^{(2)} \cdot \mu^{(3)} \ \ldots \ \mu^{(n)}} \, p_0 \, . \tag{3.4}$$

Writing equation (3.4) in recursive form yields;

$$p_n = \frac{s\mu}{\lambda^{(n)}} p_{n+1} \qquad (3.11)$$

Now apply the definitions above to substitute for $\lambda^{(n)}$;

$$p_n = \frac{s\ \mu\ p_{n+1}}{\lambda_1\omega_1(n) + \lambda_2\omega_2(n)}$$

Now insert this expression for p_n into the numerator of equation (3.10);

$$\lambda_1 \sum_{n=s}^{\infty}\{\frac{n-s+1}{s\mu}\omega_1(n)p_n\} + \lambda_2 \sum_{n=s}^{\infty}\{\frac{n-s+1}{s\mu}\omega_2(n)p_n\}$$

$$= \lambda_1 \sum_{n=s}^{\infty}\{\frac{n-s+1}{s\mu}\omega_1(n)\frac{s\ \mu\ p_{n+1}}{\lambda_1\omega_1(n) + \lambda_2\omega_2(n)}\}$$

$$+ \lambda_2 \sum_{n=s}^{\infty}\{\frac{n-s+1}{s\mu}\omega_2(n)\frac{s\ \mu\ p_{n+1}}{\lambda_1\omega_1(n) + \lambda_2\omega_2(n)}\}$$

$$= \lambda_1 \sum_{n=s}^{\infty}\{\frac{(n-s+1)\omega_1(n)p_{n+1}}{\lambda_1\omega_1(n) + \lambda_2\omega_2(n)}\}$$

$$+ \lambda_2 \sum_{n=s}^{\infty}\{\frac{(n-s+1)\omega_2(n)p_{n+1}}{\lambda_1\omega_1(n) + \lambda_2\omega_2(n)}\}$$

Letting m = n+1 yields the following expression for the numerator of equation (3.10):

$$\lambda_1 \sum_{m=s+1}^{\infty} \frac{(m-s)\,\omega_1(m-1)\,p_m}{\lambda_1\omega_1(m-1) + \lambda_2\omega_2(m-1)} + \lambda_2 \sum_{m=s+1}^{\infty} \frac{(m-s)\,\omega_1(m-1)\,p_m}{\lambda_1\omega_1(m-1) + \lambda_2\omega_2(m-1)}$$

$$= \sum_{m=s+1}^{\infty} \frac{\lambda_1\omega_1(m-1) + \lambda_2\omega_2(m-1)}{\lambda_1\omega_1(m-1) + \lambda_2\omega_2(m-1)}\,(m-s)\,p_m$$

$$= \sum_{m=s+1}^{\infty} (m-s)\,p_m \qquad (3.12)$$

Notice that equation (3.11) is an expression for the expected number of customers in the queue in steady-state. This allows us to rewrite equation (3.9) as follows:

$$E[W] = \frac{E[Q]}{\lambda_1 \sum_{n=0}^{\infty} \omega_1(n)\,p_n + \lambda_2 \sum_{n=0}^{\infty} \omega_2(n)\,p_n}$$

$$E[W] = \frac{\text{expected number in the queue}}{\text{(expected value of the rate at which customers join the system)}}$$

Thus, equation (3.10) can be seen as a generalization of Little's equation (3.2), as would be anticipated. To complete the derivation it is only necessary to insert values

for $\omega_1(n)$ and $\omega_2(n)$ as appropriate for the particular problem. In the present case, these values are

$$\omega_1(n) = 1.0 \text{ for all values of } n$$

$$\omega_2(n) = \begin{cases} 1.0 & \text{if } 0 \le n < T \\ 0.0 & \text{if } n \ge T \end{cases} .$$

Inserting these values into equation (3.9) yields

$$E[W] = \frac{E[Q]}{\lambda_1 + \lambda_2 \sum_{n=0}^{T-1} p_n}$$

which is identical to equation (3.2).

4. <u>Calculations of the Expected Waiting Time and the Probability That a Type 2 Customer is Blocked</u>

Having derived equation (3.2) as the expression for the expected waiting time, it now remains to obtain expressions for the expected number in the queue ([EQ]) and the expected rate at which customers actually join the system ($\lambda_1 + \lambda_2 \sum_{n=0}^{T-1} p_n$). Once again, these calculations are straightforward but tedious and, hence, they are presented in outline form. First consider the following expression for the expected number of customers in the queue:

$$E[Q] = \sum_{n=s}^{\infty} (n-s)p_n$$

$$= \sum_{n=s}^{\infty} np_n - s \sum_{n=s}^{\infty} p_n$$

Both of the summations above can be expressed in closed form. Details of the necessary calculations are included in Appendix A and only the results of those calculations are given below:

$$E[Q] = \frac{p_o}{s!}\{(\frac{\lambda_1+\lambda_2}{\mu})^s[\frac{s}{(1-\frac{\lambda_1+\lambda_2}{s\mu})} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{(1-\frac{\lambda_1+\lambda_2}{s\mu})^2}]$$

$$- \frac{(\lambda_1+\lambda_2)^{(T+1)}}{s^{(T+1-s)}\mu^{(T+1)}}[\frac{T+1}{(1-\frac{\lambda_1+\lambda_2}{s\mu})} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{(1-\frac{\lambda_1+\lambda_2}{s\mu})^2}]\}$$

$$+ \frac{p_o(\lambda_1+\lambda_2)^T}{s!} \frac{\lambda_1}{s^{(T+1-s)}\mu^{(T+1)}}[\frac{T+1}{(1-\frac{\lambda_1}{s\mu})} + \frac{\frac{\lambda_1}{s\mu}}{(1-\frac{\lambda_1}{s\mu})^2}]$$

$$- \{\frac{sp_o}{s!(1-\frac{\lambda_1+\lambda_2}{s\mu})}[(\frac{\lambda_1+\lambda_2}{\mu})^s - \frac{(\lambda_1+\lambda_2)^{(T+1)}}{s^{(T+1-s)}\mu^{(T+1)}}]$$

$$+ sp_o[\frac{(\lambda_1+\lambda_2)^T}{s!} \frac{\lambda_1}{s^{(T+1-s)}\mu^{(T+1)}} \frac{1}{1-\frac{\lambda_1}{s\mu}}]\}. \qquad (3.13)$$

Next, consider the expression for the expected rate at which customers actually join the system;

$$\lambda_E = \lambda_1 + \lambda_2 \sum_{n=0}^{T-1} p_n$$

$$= \lambda_1 + \lambda_2[p_o + \sum_{n=1}^{T-1} p_n]$$

$$= \lambda_1 + \lambda_2[p_o + p_o \sum_{n=1}^{s} \frac{(\lambda_1+\lambda_2)^n}{n!\mu^n}$$

$$+ p_o \sum_{n=s+1}^{T-1} \frac{(\lambda_1+\lambda_2)^n}{s!s^{(n-s)}\mu^n}] .$$

Both of the summations above were encountered in deriving the expression for p_0; the first sum cannot be expressed in closed form, but the second can and details of the calculations can be found in section two of this chapter. The final form for λ_E is:

$$\lambda_E = \lambda_1 + \lambda_2 p_0 \{1 + \sum_{n=1}^{s} \frac{(\lambda_1+\lambda_2)^n}{n!\mu^n}$$

$$+ [\frac{1}{s!(1-\frac{\lambda_1+\lambda_2}{s\mu})}(\frac{(\lambda_1+\lambda_2)^{(s+1)}}{s\mu^{(s+1)}} - \frac{(\lambda_1+\lambda_2)^T}{s^{(T-s)}\mu^T}]\} \qquad (3.14)$$

The final figure of merit is the probability that a Type 2 customer is blocked from joining the queue (P_{2b}). This is simply the probability that the number in the system equals or exceeds the queue threshold which is:

$$P_{2b} = \sum_{n=T}^{\infty} p_n = 1 - \sum_{n=0}^{T-1} p_n \, .$$

Since the summation, $\sum_{n=0}^{T-1} p_n$ was just encountered in finding λ_E it is possible to write the desired expression for P_{2b} directly:

$$P_{2b} = 1 - p_o\{1 + \sum_{n=1}^{s} \frac{(\lambda_1+\lambda_2)^n}{n!\mu^n}$$

$$+[\frac{1}{s!(1-\frac{\lambda_1+\lambda_2}{s\mu})}(\frac{(\lambda_1+\lambda_2)^{(s+1)}}{s\ \mu^{(s+1)}} - \frac{(\lambda_1+\lambda_2)^T}{s^{(T-s)}\mu^T}]\} \qquad (3.15)$$

5. Results of the Validation Efforts

At this point, expressions have been derived for all three of the figures of merit; the probability that the system is empty (p_0), the expected waiting time for any customer (E[W]), and the probability that a Type 2 customer will be blocked from joining the system (P_{2b}). The next step is to compute values for these quantities and compare the results

obtained with the values of those same quantities which result from running the computer simulation.

In order to reduce the variance in the simulation results, the method of antithetic variables was used.[5] Recall from Chapter Two that the RANDOM subroutine utilized by the program produces a string of pseudo-random numbers which range from zero to one and that it will produce the same string each time unless the "seed" number is changed. In the antithetic variable method of variance reduction two simulation runs are made for each data point desired. In the first run the random numbers are used, unmodified, to determine the sojourn time and choose the next event. In the second run, the program is modified slightly so that the random numbers are subtracted from one forming an antithetic random variable which is then used as before. Results from the two runs are then averaged and the value becomes the data point for that iteration. The seed number is then changed to produce a new random number string and the process is repeated for as many iterations as desired. Finally, the data points for all of the iterations are averaged to produce an estimate of the variable under consideration. As illustrated in Gaver and Thompson [Ref. 4], sampling by antithetic variables can reduce the variance in simulation results by

---

[5] Gaver and Thompson, pp. 584-586.

more than one-half of the variance expected using straightforward sampling techniques.

Finally, the comparison of simulation and analytical values, which serves as the validation of the simulation model, is presented in Figure 3.1. The values shown under the simulation heading were obtained by averaging over the results of ten iterations of the antithetic variable sampling technique just described; that is, twenty simulation runs were made in all. Each run simulated thirty-six hours of system operation. The comparison, measured in terms of the percent difference between the analytical and simulation values, seems to offer a good indication that the simulation model is functioning as intended. Notice that the largest discrepancy is consistently in the expected waiting time values. No explanation for this evidently systematic effect was immediately apparent.

Verification of the Simulation

Case I $\lambda_1 = 1.0$ $\lambda_2 = .5$ $\mu_1 = 1.0$ $\mu_2 = 1.0$ $\rho = .695$

	Analytical	Simulation	% Difference
E[W]	.58234	.5721195	1.75
P_0	.174387	.173633	.43
P_{2b}	.220708	.219648	.48

Case II $\lambda_1 = 1.0$ $\lambda_2 = .9$ $\mu_1 = 1.0$ $\mu_2 = 1.0$ $\rho = .799$

E[W]	.74334624	.757377	1.89
P_0	.10332954	.104199	.841
P_{2b}	.33665	.3381565	.447

Case III $\lambda_1 = 1.0$ $\lambda_2 = .99$ $\mu_1 = 1.0$ $\mu_2 = 1.0$ $\rho = .816$

E[W]	.77443362	.7874915	1.69
P_0	.09074268	.093319	2.839
P_{2b}	.36099	.3624095	.393

Case IV $\lambda_1 = 1.75$ $\lambda_2 = .249$ $\mu_1 = 1.0$ $\mu_2 = 1.0$ $\rho = .913$

E[W]	3.475469	3.58214	3.07
P_0	.043550164	.044149	1.37
P_{2b}	.69541	.695702	.042

IV. AN ANALYTICAL APPROXIMATION

A. RATIONALE FOR DERIVING AN ANALYTICAL APPROXIMATION

In the previous three chapters a simulation model was developed and verified. It was seen that the simulation appeared to do a reasonably good job of modeling the system; good enough, at least, for most engineering purposes. Why, then, go to the trouble of developing a second model? There are several reasons. First, all of the reasons for developing the simulation apply equally to an analytical model. In particular, the process of developing an analytical approximation allows an opportunity to gain additional insight into the problem from a different perspective. Such insight could lead further toward the ultimate goal, which is an exact but simple and useable analytical solution if it exists. Secondly, the implementation of the approximation described in this chapter requires tools no more sophisticated than a pocket calculator. Thus, the analytical approach offers some practical, financial advantages over the simulation. Financial advantages are even more apparent when the need to do sensitivity analysis is considered. In order to determine the sensitivity of the system to a change in one variable, it is necessary to do a large number of simulation runs for each different value of the variable. Consider Figure 3.4 in which it was necessary to do eighty simulation runs in order to be able to compare four variable changes. The

analytical approximation need only be done once for each variable change and if done on a computer, the calculations required take only a fraction of the time used by one simulation run. On the other hand, it would be easier to modify the simulation to investigate other figures of merit than it would be to write new equations for the approximation model.

B. DEVELOPMENT OF THE ANALYTICAL APPROXIMATION

Recall that in Chapter Three it was shown that an analytical solution to the problem existed in the special case in which the service rates were equal and that expressions were developed for the expected waiting time, the probability that the system is empty, and the probability that a Type 2 customer will be blocked. This suggested approximating the unequal service rate case by taking a weighted average of the service rates and using the resulting value in the equations for the equal service rate case. The intended weighting was

$$\bar{\mu} = \mu_1 \frac{\lambda_1}{\lambda_1+\lambda_2 \sum_{n=0}^{T-1} P_n} + \mu_2 \frac{\lambda_2}{\lambda_1+\lambda_2 \sum_{n=0}^{T-1} P_n} \qquad (4.1)$$

Where:

$\bar{\mu}$ = the weighted average of service rates.

All other variables are as defined previously. The weighting factors, which are the ratio of each type of customer's arrival rate to the overall system arrival rate, are intended to represent the long-run proportion of time that each operator would spend serving each type of customer. Unfortunately, this weighting scheme is infeasible since it requires values for the p_n's which are not available. Therefore, an iterative procedure was decided upon in which another type of weighting was used to get approximate values for the p_n's which could then be used in equation 4.1 to get better approximations which are again used in equation 4.1 and so on. The initial weighting utilized was

$$\bar{\mu}' = \mu_1 \frac{\lambda_1}{\lambda_1+\lambda_2} + \mu_2 \frac{\lambda_2}{\lambda_1+\lambda_2} \qquad (4.2)$$

The rationale behind this weighting scheme is the same as before, however, it can be seen that this weighting does not take into account the effect of blocking Type 2 customers.

A computer program was written which would perform the necessary calculations and iterate the procedure as many times as desired. A program listing and sample output are provided at the end of the thesis.

C. VERIFICATION OF THE ANALYTICAL APPROXIMATION

As before, the final step in developing a model is to test its performance by comparison with a suitable mathematical model. Recall that the simulation model was verified

by testing it against an analytical solution to the special case of the problem in which the two types of customers have equal service rates. This would suggest trying to find another special case of the problem in which the service rates are different and for which an exact solution could be found with which to verify the analytical approximation. Unfortunately, such a special case was not discovered. An attempt was made to solve the problem for the case of a single server with the queue threshold equal to one where Type 2 customers could only access the system when it was empty. Although it seemed that a solution should be obtainable, the complications associated with the different service rates made even this comparatively simple problem unsolvable in the time available.

Since the approximation model cannot be verified analytically, the only alternative available is to verify it by comparison to the simulation model. Of course, the simulation is itself an approximation since it is only exact in the limit as time goes to infinity and true steady-state is reached. However, it was shown in Chapter Three that, at least for small systems and equal service rates, the results obtained from running the program for thirty-six hours of simulation time differ from the exact analytical solution by less than three percent. Chapter Two described how the simulation program was able to keep track of the identity of the customer at the head of the queue and thereby correctly adjust the system service

rate when a state change occurred. Therefore, the simulation should be equally valid in the case where the service rates are different. Thus, a reasonable amount of agreement between the approximation and the simulation, while not conclusive, should be sufficient to provide confidence in the approximation model.

To see how closely the two models agree in two test cases, refer to Figure 4.1. The same method of antithetic variable sampling that was discussed in Chapter Three was used to obtain the results for the simulation model. The approximation results were obtained by carrying out ten iterations of the procedures discussed earlier in this chapter. It is interesting to note that the largest change in the figures of merit occurred between the first iteration — which utilized $\bar{\mu}' = \mu_1 \frac{\lambda_1}{\lambda_1+\lambda_2} + \mu_2 \frac{\lambda_2}{\lambda_1+\lambda_2}$ — and the second iteration, and that there was little change between succeeding iterations. In fact, there was no change at all after the fifth iteration. This effect is apparent in the sample output.

The degree of agreement between the two models shown in Figure 4.1 indicates that either model would be appropriate for a first estimate of system performance. The choice of model would therefore depend on the relative merits of each as evaluated in a particular situation. For instance, if sensitivity analysis was to be performed, the approximation model would be appropriate, while if an estimate of the

Analytical Approximation	Simulation	Percent Difference ((Sim-Anal)/(Sim)

Case I $\lambda_1 = 1.0$ $\lambda_2 = .5$ $\mu_1 = .666$ $\mu_2 = .833$

$s = 3$ $T = 6$

	Analytical Approximation	Simulation	Percent Difference
E[W]	.45110463	.4700145	4.02
p_0	.1047012	.099507	5.22
p_{2b}	.10048	.1064845	5.63

Case II $\lambda_1 = 1.75$ $\lambda_2 = .249$ $\mu_1 = 1.0$ $\mu_2 = .833$

$s = 3$ $T = 6$

	Analytical Approximation	Simulation	Percent Difference
E[W]	.39178724	.40591	3.47
p_0	.10854438	.109263	.657
p_{2b}	.11799	.12258	3.74

Figure 4.1

proportion of customers whose waiting time was greater than a given value was desired then the simulation model should be chosen.

V. CONCLUSION

Two aspects of this problem which made it interesting were the different service rates for the two types of customers and the fact that Type 2 customers are blocked from the system whenever the queue threshold is reached. Recall that Chapters One and Two discussed the implications of these aspects of the problem and showed that they made an exact analytical solution difficult, if not impossible, to achieve. In particular it was shown that the different service rates required that any solution be able to keep track of both the number and type of customers in the queue and in service as well as the identity of the customer at the head of the queue. Also, the blocking of Type 2 customers implies that the systems arrival rate is not constant but is a function of the number of customers in the system.

The only real attempt made to find an analytical solution utilized the theory of birth and death processes and it is possible that some other analytical approach could yield results. However, given the complications mentioned above, it seems that any future effort expended on this problem could best be directed toward improving the approximate solutions.

It would be worthwhile to attempt to verify both the simulation and the analytical approximation with an exact

solution to a special case with unequal service times. Chapter Four mentioned that such a verification was attempted but abandoned primarily due to time constraints. It is quite possible that additional work in this area could be successful.

The simulation algorithms should be studied to determine confidence limits on its estimates and to determine the sensitivity of the algorithm to changes in system parameters. This type of analysis would also serve to increase the confidence in the comparison between the simulation and approximation models.

Finally, there is obviously much room for improvement of the analytical approximation. Although the model developed in Chapter Four appears to perform fairly well and has an intuitive rationale, there exist a number of techniques which could be expected to yield better results. In particular the theory of Gaussian approximations and diffusion approximations discussed by Gaver and Lehoczky [Ref. 2 and 3] and Kleinrock [Ref. 6] should work well at least in the case where the system is heavily saturated.

APPENDIX A

DETAILS OF THE DERIVATION OF THE EXPRESSION FOR THE EXPECTED NUMBER IN THE QUEUE (E[Q])

$$E[Q] = \sum_{n=s}^{\infty} (n-s)p_n = \sum_{n=s}^{\infty} np_n - s \sum_{n=s}^{\infty} p_n$$

1. $$\sum_{n=s}^{\infty} np_n = \sum_{n=s}^{T} \frac{(n)(\lambda_1+\lambda_2)^n p_0}{s!s^{(n-s)}\mu^n} + \sum_{n=T+1}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^T \lambda_1{}^{n-T} p_0}{s's^{9n-s)}\mu^n}$$

a.) $$\sum_{n=s}^{T} \frac{(n)(\lambda_1+\lambda_2)^n p_0}{s!s^{(n-s)}\mu^n} = \frac{p_0}{s!}\left[\sum_{n=s}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n}\right.$$

$$\left. - \sum_{n=T+1}^{\infty} \frac{n(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n}\right]$$

1.) $$\frac{p_0}{s!} \sum_{n=s}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n} = \frac{p_0}{s!}\left(\frac{\lambda_1+\lambda_2}{\mu}\right)^s$$

$$\cdot\left[s+(s+1)\ \frac{\lambda_1+\lambda_2}{s\mu} + (s+2)\frac{(\lambda_1+\lambda_2)^2}{s^2\mu^2} + \ldots\right.$$

$$\left. + \frac{(s+k)(\lambda_1+\lambda_2)^k}{s^k\mu^k} + \ldots\ \right]$$

$$\frac{P_0}{s!} \sum_{n=s}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n}$$

$$= \frac{P_0}{s!}\left(\frac{\lambda_1+\lambda_2}{\mu}\right)^s \left[s\left(\frac{1}{1-\frac{\lambda_1+\lambda_2}{s\mu}}\right) + \sum_{n=1}^{\infty}\left(\frac{\lambda_1+\lambda_2}{s\mu}\right)^n \frac{1}{1-\frac{\lambda_1+\lambda_2}{s\mu}}\right]$$

$$= \frac{P_0}{s!}\left(\frac{\lambda_1+\lambda_2}{\mu}\right)^s \left[\frac{s}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{\left(1-\frac{\lambda_1+\lambda_2}{s\mu}\right)^2}\right]$$

2.) $$\frac{P_0}{s!} \sum_{n=T+1}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^n}{s^{(n-s)}\mu^n} = \frac{P_0}{s!} \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}$$

$$\left[(T+1)+(T+2)\ \frac{\lambda_1+\lambda_2}{s\mu} + (T+3)\left(\frac{\lambda_1+\lambda_2}{s\mu}\right)^2 + \ldots\right.$$

$$\left. + (T+k)\left(\frac{\lambda_1+\lambda_2}{s\mu}\right)^{k-1} + \ldots\right]$$

$$= \frac{P_0}{s!} \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}\left[(T+1)\ \frac{1}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{\left(1-\frac{\lambda_1+\lambda_2}{s\mu}\right)^2}\right]$$

$$\sum_{n=s}^{T} \frac{(n)(\lambda_1+\lambda_2)^n P_0}{s!\,s^{(n-s)}\mu^n} = \frac{P_0}{s!}\left\{\left(\frac{\lambda_1+\lambda_2}{\mu}\right)^2\left[\frac{s}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{\left(1-\frac{\lambda_1+\lambda_2}{s\mu}\right)^2}\right]\right.$$

$$\left. - \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}\left[\frac{T+1}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{\left(1-\frac{\lambda_1+\lambda_2}{s\mu}\right)^2}\right]\right\}$$

b.) $$\sum_{n=T+1}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^T \lambda_1^{n-T} p_0}{s!s^{(n-s)}\mu^n}$$

$$= \frac{p_0(\lambda_1+\lambda_2)^T}{s!} \sum_{n=T+1} \frac{n\lambda_1^{n-T}}{s^{(n-s)}\mu^n}$$

$$= \frac{p_0(\lambda_1+\lambda_2)^T}{s!} \frac{\lambda_1}{s^{T+1-s}\mu^{T+1}}[(T+1)+(T+2)\frac{\lambda_1}{s\mu}$$

$$+ (T+1)(\frac{\lambda_1}{s\mu})^2 + \ldots]$$

$$\sum_{n=T+1}^{\infty} \frac{(n)(\lambda_1+\lambda_2)^T \lambda_1^{n-T} p_0}{s!s^{(n-s)}\mu^n} = \frac{p_0(\lambda_1+\lambda_2)^T}{s!} \frac{\lambda_1}{s^{T+1-s}\mu^{T+1}}[(T+1)\frac{1}{1-\frac{\lambda_1}{s\mu}}$$

$$+ \frac{\frac{\lambda_1}{s\mu}}{(1-\frac{\lambda_1}{s\mu})^2}]$$

$$\sum_{n=s}^{\infty} np_n = \frac{p_0}{s!}\{(\frac{\lambda_1+\lambda_2}{\mu})^s[\frac{s}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{(1-\frac{\lambda_1+\lambda_2}{s\mu})^2}]$$

$$- \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}[\frac{T+1}{1-\frac{\lambda_1+\lambda_2}{s\mu}} + \frac{\frac{\lambda_1+\lambda_2}{s\mu}}{(1-\frac{\lambda_1+\lambda_2}{s\mu})^2}]\}$$

$$+ \frac{p_0(\lambda_1+\lambda_2)}{s!} \frac{\lambda_1}{s^{T+1-s}\mu^{T+1}}[\frac{T+1}{1-\frac{\lambda_1}{s\mu}} + \frac{\frac{\lambda_1}{s\mu}}{(1-\frac{\lambda_1}{s\mu})^2}]$$

$$2.\quad s\sum_{n=s}^{\infty} p_n = s\Big[\sum_{n=s}^{T} \frac{(\lambda_1+\lambda_2)^n p_0}{s!\,s^{(n-s)}\mu^n}$$

$$+\sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^T \lambda_1^{\,n-T} p_0}{s!\,s^{(n-s)}\mu^n}\Big] \qquad \text{(A.1)}$$

$$\text{a.)}\quad s\sum_{n=s}^{T} \frac{(\lambda_1+\lambda_2)^n p_0}{s!\,s^{(n-s)}\mu^n}$$

$$= \frac{sp_0}{s!}\Big[\frac{(\lambda_1+\lambda_2)^s}{\mu^s}\sum_{j=0}^{\infty}\Big(\frac{\lambda_1+\lambda_2}{s\mu}\Big)^j - \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}\sum_{j=0}^{\infty}\Big(\frac{\lambda_1+\lambda_2}{s\mu}\Big)^j\Big]$$

$$= \frac{s\,p_0}{s!\,\big(1-\frac{\lambda_1+\lambda_2}{s\mu}\big)}\Big[\Big(\frac{\lambda_1+\lambda_2}{\mu}\Big)^s - \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}\Big]$$

$$\text{b.)}\quad s\sum_{n=T+1}^{\infty} \frac{(\lambda_1+\lambda_2)^T \lambda_1^{\,n-T}\, p_0}{s!\,s^{(n-s)}\mu^n}$$

$$= sp_0\Big[\frac{(\lambda_1+\lambda_2)^T}{s!}\Big(\frac{\lambda_1}{s^{T+1-s}\mu^{T+1}}\Big)\Big(\frac{1}{1-\frac{\lambda_1}{s\mu}}\Big)\Big]$$

$$s\sum_{n=s}^{\infty} p_n = \frac{s\,p_0}{s!\,\big(1-\frac{\lambda_1+\lambda_2}{s\mu}\big)}\Big[\Big(\frac{\lambda_1+\lambda_2}{\mu}\Big)^s - \frac{(\lambda_1+\lambda_2)^{T+1}}{s^{T+1-s}\mu^{T+1}}\Big]$$

$$+\,p_0\Big[\frac{(\lambda_1+\lambda_2)^T}{s!}\Big(\frac{\lambda_1}{s^{T+1-s}\mu^{T+1}}\Big)\Big(\frac{1}{1-\frac{\lambda_1}{s\mu}}\Big)\Big] \qquad \text{(A.2)}$$

Finally, combining A.1 and A.2 yields the result shown in Chapter 3.

PROGRAM LISTING FOR SIMULATION ALGORITHM

```
C       THIS PROGRAM SIMULATES A MULTI-SERVER QUEUING SYSTEM
C  WHICH IS BEING ACCESSED BY TWO TYPES OF CUSTOMERS. THE TWO
C  TYPES OF CUSTOMERS HAVE DIFFERENT ARRIVAL AND SERVICE
C  RATES. TYPE 1 CUSTOMERS ARE ALLOWED TO JOIN THE SYSTEM RE-
C  GARDLESS OF QUEUE LENGTH. TYPE 2 CUSTOMERS ARE BLOCKED
C  FROM THE QUEUE WHEN THE QUEUE LENGTH EXCEEDS A GIVEN THRES
C  HOLD. THE SIMULATION IS BASED ON RANDOMLY CHOSEN SOJOURN
C  TIMES BETWEEN EVENTS WITH THE NEXT EVENT TO OCCUR ALSO
C  CHOSEN RANDOMLY.
C       THE VARIABLES USED IN THE PROGRAM ARE DEFINED BELOW.
C            LAMDA1=ARRIVAL RATE OF TYPE 1 CUSTOMERS
C            LAMDA2=ARRIVAL RATE OF TYPE 2 CUSTOMERS
C            MU1=SERVICE RATE OF TYPE 1 CUSTOMERS
C            MU2=SERVICE RATE OF TYPE 2 CUSTOMERS
C            KQ=THE QUEUE THRESHOLD FOR BLOCKING OF TYPE 2
C            KS=THE NUMBER OF SERVERS
C            J1=THE NUMBER OF TYPE 1 CUSTOMERS IN SERVICE
C            J2=THE NUMBER OF TYPE 2 CUSTOMERS IN SERVICE
C            NUM1=THE NUMBER OF TYPE 1 ARRIVALS
C            NUM2=THE NUMBER OF TYPE 2 ARRIVALS
C       NBLOCK=THE NUMBER OF TYPE 2 CUSTOMERS BLOCKED.
C            I=THE NUMBER OF BOTH TYPES WHO HAVE JOINED THE QUE
C            J=THE NUMBER OF BOTH TYPES WHO HAVE LEFT THE QUEUE
C            SOJTIM=THE SOJOURN TIME=AN EXPONENTIAL RANDOM VARI
C                 1/(LAMDA1+LAMDA2+J1*MU1+J2*MU2)
      REAL*4 LAMDA1,LAMDA2,MU1,MU2
      COMMON LAMDA1,LAMDA2,TIME,STIME,QTIME,KQ,KS,IX,J1,J2,N
     1UMQ1,NUMQ2,NUM1,NUM2,NBLOCK,I,J,NOWAIT,NEXT1
      DIMENSION U(2),NEXT1(2000),QTIME(2000),STIME(2000),WTI
     1ME(2000)
      READ(5,5) LAMDA1,LAMDA2,MU1,MU2,TMAX
    5 FORMAT(5F10.5)
      READ(5,10) J1,J2,NUMQ1,NUMQ2,KSTOP,KS,KQ
   10 FORMAT(7I5)
      WRITE(6,12) KSTOP
12    FORMAT(T45,'NUMBER OF ITERATIONS THIS RUN',I4//)
      WRITE(6,15) LAMDA1,LAMDA2,MU1,MU2,TMAX,KS,KQ
15    FORMAT(T45,'INITIAL CONDITIONS',//,T20,'TYPE 1 ',
     1'ARRIVAL RATE',T50,F10.5,/,T20,'TYPR 2 ARRIVAL RATE',T
     150,F10.5,/,T20,'TYPE 1 SERVICE RATE',T50,F10.5,/,T20,
     1'TYPE 2 SERVICE RATE',T50,F10.5,/,T20,'MAX.SIMULATI',
     1'ON TIME',T50,F10.5,/,T20,'NUMBER OF SERVERS',T50,I8,/
     1,T20,'QUEUE THRESHOLD',T50,I8//)
      WRITE(6,20) J1,J2,NUMQ1,NUMQ2
20    FORMAT(T45,'INITIAL STATE VARIABLES',//,T20,'NUMBER ',
     1'OF TYPE 1 IN SERVICE',T50,I8,/,T20,'NUMBER OF TYPE ',
     1'2 IN SERVICE',T50,I8,/,T20,'NUMBER OF TYPE 1 IN ',
     1'QUEUE',T50,I8,/,T20,'NUMBER OF TYPE 2 IN QUEUE',T50,I
     18//)
      CALL OVFLOW
      SAVE2=LAMDA2
      IF(ITEM.EQ.1) GO TO 27
      J1=0
      J2=0
      NUMQ1=0
      NUMQ2=0
27    CUMWT=0.0
      KUMNQD=0
      KUMIN=0
      TIME=0.0
      ZERTIM=0.0
      BLKTIM=0.0
      I=0
      J=0
      DO 500 K=1,KSTOP
      NUM1=0
```

```
      NUM2=0
      NBLOCK=0
      NOWAIT=0
      NLINQ=NUMQ1+NUMQ2
30    K2=NUMQ1+ NUMQ2
      IF(K2.GE.KQ) GO TO 3000
      LAMDA2=SAVE2
      GO TO 3001
3000  LAMDA2=0.0
3001  D=LAMDA1+ LAMDA2+J1*MU1+J2*MU2

C     THE FOUR POSSIBLE EVENTS ARE TYPE 1 ARRIVAL, TYPE 2
C ARRIVAL,TYPE 1 SERVICE AND TYPE 2 SERVICE. THE EVENTS ARE
C CHOSEN RANDOMLY AND THE NECESSARY CHANGES IN THE STATE
C VARIABLES J1,J2,NUMQ1,NUMQ2 ARE MADE BY CALLING THE APPRO-
C PRIATE SUBROUTINE.

      CALL RANDOM(IX,U,3)
      SOJTIM=(-1.0/D)*ALOG(U(1))
      IF(K2.LT.KQ) GO TO 3002
      BLKTIM=BLKTIM+SOJTIM
      T2INT=(-1.0/SAVE2)*ALOG(U(3))
      IF(T2INT.LE.SOJTIM) NBLOCK=NBLOCK+1
3002  TIME=TIME+SOJTIM
      KSYS=J1+J2+NUMQ1+NUMQ2
      IF(KSYS.GT..1) GO TO 3003
      ZERTIM=ZERTIM+SOJTIM
C
C
3003  IF(U(2).LE.LAMDA1/D) GO TO 31
      IF(U(2).LE.(LAMDA1+LAMDA2)/D) GO TO 32
      IF(U(2).LE.(LAMDA1+LAMDA2+J1*MU1)/D) GO TO 33
      IF(J2.EQ.0) GO TO 41
      CALL ROUTE4
      GO TO 49
31    CALL ROUTE1
      GO TO 49
32    CALL ROUTE2
      GO TO 49
   33 IF(J1.EQ.0) GO TO 41
      CALL ROUTE3
      GO TO 49
   41 WRITE(6,42)
   42 FORMAT(T55,'ERROR.SERVICE ATTEMPTED WITH 0 CUSTOMERS')
      STOP
49    FK=FLOAT(K)
      IF(TIME.LE.FK*TMAX) GO TO 30
      WRITE(6,50) K,TIME
50    FORMAT(//T45,'END OF INTERVAL',I3,/,T45,'CUMULATIVE ',
     1'TIME IS',F12.5)
C
C
C     THE FOLLOWING BLOCK OF STATEMENTS (1)COMPUTES THE NUM-
C BER WHO QUEUED (2)COMPUTES THE WAITING TIME FOR EACH
C (WTIME(M)) AND (3)COMPUTES THE AVERAGE WAITING TIME (CUM-
C AVG).
      NQD=I-NLINQ
      NLINQ=NUMQ1+NUMQ2
      WTSUM=0.0
      DO 80 M=1,J
      WTIME(M)=STIME(M)-QTIME(M)
      WTSUM=WTSUM+WTIME(M)
   80 CONTINUE
      CUMWT=CUMWT+WTSUM
      KUMNQD=KUMNQD+I
      KUMIN=KUMIN+NUM1+NUM2
      CUMAVG=CUMWT/(KUMIN-(I-J))
      WRITE(6,90) CUMAVG
90    FORMAT(/T20,'CUMULATIVE AVERAGE WAITING TIME FOR ALL '
     1,'CUSTOMERS',/,T20,'EXCEPT THOSE BLOCKED',F12.5)
      PO=ZERTIM/TIME
      WRITE(6,92) PO
```

```
92    FORMAT(//T20,'CUMULATIVE PROPORTICN CF TIME THAT N=0',
     1F12.5)
      PKQ=BLKTIM/TIME
      WRITE(6,93) PKC
93    FORMAT(//T20,'CUMULATIVE PROPORTICN CF TIME THAT TYPE'
     1,' 2 CUSTOMERS',/,T20,'WOULD BE BLOCKED',F12.5)
      IF((I-J).EQ.0) GO TO 250
      ISTOP=I-J
      DO 200 L=1,ISTOP
      CTIME(L)=CTIME(J+L)
200   CONTINUE
250   I=I-J
      J=0
  500 CONTINUE
      STOP
      END
```

SUBROUTINE LISTINGS FOR SIMULATION ALGORITHM

```
      SUBROUTINE ROUTE1
C     THIS SUBROUTINE IS CALLED IN EVENT OF A TYPE 1 ARRIVAL
      REAL*4 LAMDA1,LAMDA2,MU1,MU2
      COMMON LAMDA1,LAMDA2,TIME,STIME,QTIME,KQ,KS,IX,J1,J2,N
     1UMQ1,NUMQ2,NUM1,NUM2,NBLOCK,I,J,NOWAIT,NEXT1
      DIMENSION U(2),NEXT1(2000),QTIME(2000),STIME(2000),WTI
     1ME(2000)
      NUM1=NUM1+1
      IF(J1+J2.EQ.KS) GO TO 10
      J1=J1+1
      NOWAIT=NOWAIT+1
      GO TO 20
   10 NUMQ1=NUMQ1+1
      I=I+1
      NEXT1(I)=1
      QTIME(I)=TIME
   20 RETURN
      END


      SUBROUTINE ROUTE2
C     THIS SUBROUTINE IS CALLED IN EVENT OF A TYPE 2 ARRIVAL
      REAL*4 LAMDA1,LAMDA2,MU1,MU2
      COMMON LAMDA1,LAMDA2,TIME,STIME,QTIME,KQ,KS,IX,J1,J2,N
     1UMQ1,NUMQ2,NUM1,NUM2,NBLOCK,I,J,NOWAIT,NEXT1
      DIMENSION U(2),NEXT1(2000),QTIME(2000),STIME(2000),WTI
     1ME(2000)
      NUM2=NUM2+1
      IF(J1+J2.EQ.KS) GO TO 20
      J2=J2+1
      NOWAIT=NOWAIT+1
      GO TO 30
   20 NUMQ2=NUMQ2+1
      I=I+1
      NEXT1(I)=2
      QTIME(I)=TIME
   30 RETURN
      END


      SUBROUTINE ROUTE3
C     THIS SUBROUTINE IS CALLED IN EVENT OF A TYPE 1 SERVICE
      REAL*4 LAMDA1,LAMDA2,MU1,MU2
      COMMON LAMDA1,LAMDA2,TIME,STIME,QTIME,KQ,KS,IX,J1,J2,N
     1UMQ1,NUMQ2,NUM1,NUM2,NBLOCK,I,J,NOWAIT,NEXT1
      DIMENSION U(2),NEXT1(2000),QTIME(2000),STIME(2000),WTI
```

```
     1ME(2000)
      KQUE=NUMQ1+NUMQ2
      IF(KQUE.GT..1) GO TO 10
      J1=J1-1
      GO TO 30
   10 J=J+1
      IF(NEXT1(J).GT.1) GO TO 20
      NUMQ1=NUMQ1-1
      GO TO 25
   20 J1=J1-1
      J2=J2+1
      NUMQ2=NUMQ2-1
   25 STIME(J)=TIME
   30 RETURN
      END


      SUBROUTINE ROUTE4
C     THIS SUBROUTINE IS CALLED IN EVENT OF A TYPE 2 SERVICE
      REAL*4 LAMDA1,LAMDA2,MU1,MU2
      COMMON LAMDA1,LAMDA2,TIME,STIME,QTIME,KQ,KS,IX,J1,J2,N
     1UMQ1,NUMQ2,NUM1,NUM2,NBLOCK,I,J,NCWAIT,NEXT1
      DIMENSION U(2),NEXT1(2000),QTIME(2000),STIME(2000),WTI
     1ME(2000)
      KQUE=NUMQ1+NUMQ2
      IF(KQUE.GT..1) GO TO 10
      J2=J2-1
      GO TO 30
   10 J=J+1
      IF(NEXT1(J).GT.1) GO TO 20
      NUMQ1=NUMQ1-1
      J1=J1+1
      J2=J2-1
      GO TO 25
   20 NUMQ2=NUMQ2-1
   25 STIME(J)=TIME
   30 RETURN
      END
```

PROGRAM LISTING FOR THE ANALYTICAL APPROXIMATION

```
C     THIS DOUBLE PRECISION PROGRAM COMPUTES (1) A WEIGHTED
C AVERAGE SYSTEM SERVICE RATE (DMU),(2)THE PROBILITY THAT
C THE SYSTEM IS EMPTY (PO),(3)THE EXPECTED NUMBER OF CUSTO-
C MERS IN THE QUEUE (EQ),AND (4)THE EXPECTED WAITING TIME
C FOR ANY CUSTOMER (EW). THE GENERAL EQUATIONS DEVELOPED IN
C CHAPTER 2 OF THE THESIS ARE USED IN AN ITERATIVE PROCEDURE
C AS DESCRIBED IN CHAPTER 4. TO IMPROVE READABILITY, THE EQ-
C UATIONS ARE BROKEN DOWN INTO A NUMBER OF TERMS (TERM1 THRU
C TERM8).

      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DMU(20),PO(20),PSUM(20),DLEFF(20),EQ(20),
     1EW(20),P2B(20)
      READ(2,13)DL1,DL2,U1,U2
13    FORMAT(4D15.8)
      WRITE(6,50)DL1,DL2,U1,U2
50    FORMAT(4D15.8)
      READ(2,18) KS,KT,IMAX
18    FORMAT(3I6)
      WRITE(6,55)KS,KT,IMAX
55    FORMAT(3I6)
      DO 1000 I=1,IMAX
      IF(I.GT.1) GO TO 20
      DMU(I)=(U1*DL1+U2*DL2)/(DL1+DL2)
      GO TO 25
20    DMU(I)=(U1*DL1+U2*DL2*PSUM(I-1))/(DL1+DL2*PSUM(I-1))
25    SUM1=0.0D0
          DO 100 J=1,KS
          SUM1=SUM1+(DL1+DL2)**J/(FACT(J)*DMU(I)**J)
100       CONTINUE
      F1=1.0D0-(DL1+DL2)/(KS*DMU(I))
      F2=(DL1+DL2)/(KS*DMU(I))
      F3=1.0D0-DL1/(KS*DMU(I))
      TERM1=1.0D0/(FACT(KS)*F1)
      TERM2=((DL1+DL2)**(KS+1)/(KS*DMU(I)**(KS+1)))-
     1((DL1+DL2)**(KT+1)/(KS**(KT+1-KS)*DMU(I)**(KT+1)))
      TERM3=((DL1+DL2)**KT*DL1)/(FACT(KS)*KS**(KT+1-KS)*
     1DMU(I)**(KT+1)*(1.0D0-DL1/(KS*DMU(I))))
      PO(I)=1.0D0/(1.0D0+SUM1+TERM1*TERM2+TERM3)
      TERM4=((DL1+DL2)**(KS+1)/(KS*DMU(I)**(KS+1)))-
     1((DL1+DL2)**KT/(KS**(KT-KS)*DMU(I)**KT))
      PSUM(I)=PO(I)*(1.0DO+SUM1+TERM4/(FACT(KS)*F1))
      DLEFF(I)=DL1+DL2*PSUM(I)
      TERM5=((DL1+DL2)/DMU(I))**KS*(KS/F1+F2/F1**2)
      TERM6=((DL1+DL2)**(KT+1)/(KS**(KT+1-KS)*DMU(I)**
     1(KT+1)))*((KT+1)/F1+F2/F1**2)
      TERM7=((PO(I)*(DL1+DL2)**KT*DL1)/(FACT(KS)*KS**
     1(KT+1-KS)*DMU(I)**(KT+1)))*((KT+1)/F3+(DL1/
     1(KS*DMU(I)))/F3**2)
      TERM8=(((DL1+DL2)/DMU(I))**KS-(DL1+DL2)**(KT+1)/
     1(KS**(KT+1-KS)*DMU(I)**(KT+1)))/(FACT(KS)*F1)
      EQ(I)=(PO(I)/FACT(KS))*(TERM5-TERM6)+TERM7-KS*PO(I)*
     1(TERM8+TERM3)
      EW(I)=EQ(I)/DLEFF(I)
      P2B(I)=1.0D0-PSUM(I)
      WRITE(6,200)DMU(I),PO(I),PSUM(I),DLEFF(I),EW(I),P2B(I)
200   FORMAT(//T5,D15.8,T26,D15.8,T46,D15.8,T66,D15.8,T86,
     1D15.8,T106,D12.5)
1000  CONTINUE
      STOP
      END


      DOUBLE PRECISION FUNCTION FACT(N)
C     THIS SUBROUTINE USES A GAMMA FUNCTION TO COMPUTE THE
C  FACTORIAL OF THE INTEGER N.
      IMPLICIT REAL*8(A-H,O-Z)
```

```
      IF(N.GT.1) GO TO 10
      FAC1=1.0D0
      RETURN
10    K=N+1
      DN=DFLOAT(K)
      FAC1=DGAMMA(DN)
      RETURN
      END
```

SAMPLE OUTPUT FOR ANALYTICAL APPROXIMATION

```
SYSTEM PARAMETERS

TYPE 1 ARRIVAL RATE 0.17500000D 01
TYPE 2 ARRIVAL RATE 0.24900000D 00
TYPE 1 SERVICE RATE 0.10000000D 01
TYPE 2 SERVICE RATE 0.83300000D 00
NUMBER OF SERVERS  3
QUEUE THRESHOLD  6
NUMBER OF ITERATIONS 10

ITERATION NUMBER  1

THE WEIGHTED AVG. SERVICE RATE IS 0.97919810D 00
THE VALUE OF PO IS 0.10787409D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19693122D 01
THE EXPECTED WAITING TIME IS 0.39563682D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11922827D 00

ITERATION NUMBER  2

THE WEIGHTED AVG. SERVICE RATE IS 0.98140207D 00
THE VALUE OF PO IS 0.10855151D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19696232D 01
THE EXPECTED WAITING TIME IS 0.39174657D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11797899D 00

ITERATION NUMBER  3

THE WEIGHTED AVG. SERVICE RATE IS 0.98137863D 00
THE VALUE OF PO IS 0.10854430D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19696199D 01
THE EXPECTED WAITING TIME IS 0.39178767D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11799220D 00

ITERATION NUMBER  4

THE WEIGHTED AVG. SERVICE RATE IS 0.98137888D 00
THE VALUE OF PO IS 0.10854438D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19696200D 01
THE EXPECTED WAITING TIME IS 0.39178723D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11799206D 00

ITERATION NUMBER  5

THE WEIGHTED AVG. SERVICE RATE IS 0.98137888D 00
THE VALUE OF PO IS 0.10854438D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19696200D 01
THE EXPECTED WAITING TIME IS 0.39178724D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11799206D 00

ITERATION NUMBER  6

THE WEIGHTED AVG. SERVICE RATE IS 0.98137888D 00
THE VALUE OF PO IS 0.10854438D 00
THE EFFECTIVE SYSTEM ARRIVAL RATE IS 0.19696200D 01
THE EXPECTED WAITING TIME IS 0.39178724D 00
THE PROB. THAT TYPE TWOS ARE BLOCKED IS 0.11799206D 00
```

SAMPLE OUTPUT FOR SIMULATION ALGORITHM

```
                                NUMBER OF ITERATIONS THIS RUN  36

                                INITIAL CONDITIONS

TYPE 1 ARRIVAL RATE                    1.75000
TYPR 2 ARRIVAL RATE                    0.24900
TYPE 1 SERVICE RATE                    1.00000
TYPE 2 SERVICE RATE                    0.83300
MAX.SIMULATION TIME                   60.00000
NUMBER OF SERVERS                            3
QUEUE THRESHOLD                              3

                                INITIAL STATE VARIABLES

NUMBER OF TYPE 1 IN SERVICE                  1
NUMBER OF TYPE 2 IN SERVICE                  1
NUMBER OF TYPE 1 IN QUEUE                    0
NUMBER OF TYPE 2 IN QUEUE                    0


                                END OF INTERVAL  1
                                CUMULATIVE TIME IS     63.03427

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED     -7.11888

CUMULATIVE PROPORTION OF TIME THAT N=0      0.13202

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED      0.55560 .

                                END OF INTERVAL  2
                                CUMULATIVE TIME IS    122.57188

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED    -14.23775

CUMULATIVE PROPORTION OF TIME THAT N=0      0.06789

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED      0.77146
```

END OF INTERVAL 33
CUMULATIVE TIME IS 1982.23901

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED -234.92163

CUMULATIVE PROPORTION OF TIME THAT N=0 0.00420

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED 0.98587

END OF INTERVAL 34
CUMULATIVE TIME IS 2041.77637

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED -242.04044

CUMULATIVE PROPORTION OF TIME THAT N=0 0.00408

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED 0.98628

END OF INTERVAL 35
CUMULATIVE TIME IS 2101.31372

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED -249.15926

CUMULATIVE PROPORTION OF TIME THAT N=0 0.00396

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED 0.98667

END OF INTERVAL 36
CUMULATIVE TIME IS 2160.85107

CUMULATIVE AVERAGE WAITING TIME FOR ALL CUSTOMERS
EXCEPT THOSE BLOCKED -256.27808

CUMULATIVE PROPORTION OF TIME THAT N=0 0.00385

CUMULATIVE PROPORTION OF TIME THAT TYPE 2 CUSTOMERS
WOULD BE BLOCKED 0.98704

LIST OF REFERENCES

1. Cooper, R.B., Introduction to Queuing Theory, McMillan, 1972.

2. Naval Postgraduate School Report 55-77-16, A Diffusion Approximation Analysis of a General n-Compartment System by D.P. Gaver and J.P. Lehoczky, April 1977.

3. Gaver, D.P., and Lehoczky, J.P., Gaussian Approximation to Service Problems: A Communication System Example, to appear.

4. Gaver, D.P. and Thompson, G.L., Programming and Probability Models in Operations Research, Brooks/Cole, 1973.

5. Hillier, F.S., and Lieberman, G.J., Introduction to Operations Research, Holden-Day, 1967.

6. Kleinrock, L., Queuing Systems Volume II: Computer Applications, Wiley and Sons, 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginai 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor Donald P. Gaver, Code 55Gv Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4.	Professor M.G. Sovereign, Code 55Zo Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
5.	Capt. Joseph F. Jennings USMC c/o Marine Corps Liaison Officer U.S. Army Armor Center Ft. Knox, Kentucky	1